CLAIMS

What is claimed is:

1. A method for filtering transport layer connections with application layer information, comprising the steps of:

receiving a connection request having an application layer component and a transport layer component;

providing a connection database to store information about connection requests and associated application layer outcomes;

providing a throttle filter using data from the connection database, the throttle filter to filter the connection request at the transport layer component;

applying the throttle filter to the received connection request;

if the throttle filter blocks the transport layer component of the connection request, dropping the connection request silently; and

if the throttle filter allows the transport layer component of the connection request, proceeding with the application layer component.

2. The method of claim 1 further comprising the steps of:

adding data from an application layer outcome of the connection request to the connection database; and

updating the throttle filter with information from the connection database.

3. The method of claim 2 wherein the step of adding data comprises the steps of:

recording a connection requestor identifier to the connection database; and

providing a connection requestor rank to the connection requestor identifier based on an outcome of the application layer connection component.

4. The method of claim 2 wherein the step of updating the throttle filter with information from the connection database comprises periodically replacing throttle filter data with a preselected number of connection requestor identifiers ranked least desirable in the connection database.

5. The method of claim 1 wherein the throttle filter is a list of connection request characteristics and the step of applying the throttle filter further comprises comparing data from the connection request to the list of connection request characteristics.

6. The method of claim 5 wherein the list of connection request characteristics further comprises a list of connection requestor IP addresses to be blocked as indicated by data from the connection database.

7. The method of claim 5 wherein the list of connection request characteristics further comprises a list of connection requestor port numbers to be blocked as indicated by data from the connection database.

8. The method of claim 5 wherein the list of connection request characteristics further comprises a list of connection requestor virtual routing forwarding table IDs to be blocked as indicated by data from the connection database.

9. The method of claim 1 wherein the step of applying the throttle filter further comprises the steps of:

determining whether a limit of connections created in a connection cycle period has been exceeded;

if the limit of connections created has been exceeded, dropping the connection request;

if the limit of connections created has not been exceeded, determining whether a rate of incoming connections has been exceeded;

if the rate of incoming connections has been exceeded, then dropping the connection request; and

if the rate of incoming connections has not been exceeded, then comparing requestor identification information in the connection request to data in the throttle filter.

10. The method of claim 1 wherein the connection request is an HTTP request, the application layer component is an HTTP connection component and the transport layer component is TCP connection component.

11. The method of claim 1 wherein the connection request is an HTTPS request, the application layer component is an HTTPS connection component and the transport layer component is TCP connection component.

12. A system to filter server connections in an embedded system, comprising:

a network interface to receive a connection request from a requestor, the connection request having an application layer connection component and a transport layer connection component;

a filter device to filter connections using the transport layer connection component, the filter device including a connection database and a throttle filter, the connection database to store information about connection requests and application layer connection component outcomes, the throttle filter having data from the connection database to filter connection requests using the transport layer connection component; and

a controller coupled to the filter device and the network interface, the controller to apply the throttle filter to the transport layer connection component of the connection request, to drop the connection request silently if the throttle filter blocks the transport layer component, to proceed with an application layer connection if the throttle filter allows the transport layer component, to add data about the application layer connection to the connection database, and to update the throttle filter with information about the connection database.

13. The system of claim 12 wherein the server connection is an HTTP server connection, the application layer connection component is an HTTP connection component, and the transport layer connection component is a TCP connection component.

14. The system of claim 12 wherein the server connection is an HTTPS server connection, the application layer connection component is an HTTPS connection component, and the transport layer connection component is a TCP connection component.

15. The system of claim 12 wherein the filter device further comprises a rate limiter to switch the filter device between global and selective modes, the rate limiter to switch the filter device to global mode if a rate limit threshold is exceeded and to switch the filter device to selective mode if the rate limit threshold is not exceeded; and

the controller configured to drop the connection request silently without applying the throttle filter if the filter device is in global mode and to apply the throttle filter if the filter device is in selective mode.

16. The system of claim 15 wherein the rate limit threshold further comprises a limit of connections created in a connection cycle period.

17. The system of claim 15 wherein the rate limit threshold further comprises a rate of incoming connections.

18. The system of claim 12 wherein the connection database is a table in which each entry has an IP address of a connection requestor and an associated rank based on an outcome of a connection attempted in response to a connection request from the connection requestor.

19. The system of claim 18 wherein each entry of the table further includes a port number of the connection requestor.

20. The system of claim 18 wherein each entry of the table further includes a virtual routing forwarding table ID of the connection requestor.

21. The system of claim 12 wherein each entry in the table includes an entry age, the filter device configured to delete entries having an entry age that exceeds an age threshold.

22. The system of claim 12 wherein the throttle filter is a list of IP addresses of connection requestors to be blocked as indicated by data from the connection database.

23. The system of claim 22 wherein the throttle filter further includes port numbers of connection requestors to be blocked as indicated by data from the connection database.

24. The system of claim 22 wherein the throttle filter further includes virtual routing forwarding table IDs of connection requestors to be blocked as indicated by data from the connection database.

25. A method for filtering HTTP server connections in an embedded system, comprising the steps of:

receiving a connection request having an HTTP connection component and a TCP connection component;

providing a connection database to store information about connection requests and associated HTTP connection outcomes;

providing a throttle filter using data from the connection database, the throttle filter to filter the connection request at the TCP connection component;

determining whether a limit of connections created in a connection cycle period has been exceeded;

if the limit of connections created has been exceeded, dropping the connection request silently;

if the limit of connections created has not been exceeded, determining whether a rate of incoming connections has been exceeded;

if the rate of incoming connections has been exceeded, then dropping the connection request silently;

if the rate of incoming connections has not been exceeded, then comparing requestor identification information in the TCP connection component of the connection request to data in the throttle filter;

if the throttle filter blocks the TCP connection component, dropping the connection request silently;

if the throttle filter allows the TCP connection component, proceeding with the HTTP connection component;

adding data from the HTTP connection component to the connection database; and

updating the throttle filter with information from the connection database.

26. A method for filtering HTTPS server connections in an embedded system, comprising the steps of:

receiving a connection request having an HTTPS connection component and a TCP connection component;

providing a connection database to store information about connection requests and associated HTTPS connection outcomes;

providing a throttle filter using data from the connection database, the throttle filter to filter the connection request at the TCP connection component;

determining whether a limit of connections created in a connection cycle period has been exceeded;

if the limit of connections created has been exceeded, dropping the connection request silently;

if the limit of connections created has not been exceeded, determining whether a rate of incoming connections has been exceeded;

if the rate of incoming connections has been exceeded, then dropping the connection request silently;

if the rate of incoming connections has not been exceeded, then comparing requestor identification information in the TCP connection component of the connection request to data in the throttle filter;

if the throttle filter blocks the TCP connection component, dropping the connection request silently;

if the throttle filter allows the TCP connection component, proceeding with the HTTPS connection component;

adding data from the HTTPS connection component to the connection database; and

updating the throttle filter with information from the connection database.

27. A computer program product having a computer-readable medium including computer program logic encoded thereon that, when performed on a computer system directs the computer system to perform the method of:

receiving a connection request having an application layer component and a transport layer component;

providing a connection database to store information about connection requests and associated application layer outcomes;

providing a throttle filter using data from the connection database, the throttle filter to filter the connection request at the transport layer component;

applying the throttle filter to the received connection request;

if the throttle filter blocks the transport layer component of the connection request, dropping the connection request silently; and

if the throttle filter allows the transport layer component of the connection request, proceeding with the application layer component.